



UNIVERSITY OF GOTHENBURG



Improving Product Development Process & Quality using Electronic Checklists

- A case study

Master of Science Thesis in the Programme Software Engineering and Management

ABDULLAH ARSLAN
ZHE LI

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
Göteborg, Sweden, August 2011

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.

The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Improving Product Development Process and Quality using Electronic Checklist Case Study

ABDULLAH ARSLAN
ZHE LI

© ABDULLAH ARSLAN, June 2011.

© ZHE LI, June 2011.

Examiner: DAG HENRIK BERGSJÖ

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
Göteborg, Sweden June 2011

ABSTRACT

Knowledge has been an important asset for organizations. To be successful, organizations should manage their organization knowledge efficiently. An important part of organizational knowledge lies as “experience” in tacit form. Especially in engineering companies, that tacit knowledge could be much valuable to discard. A company can only benefit from that tacit knowledge by converting it to some “transferrable form”. Kokkonen uses “experience knowledge” for the knowledge that is sourced from experience in company’s operations and captured in some medium. The level that a company collects and utilizes its experience knowledge has an important effect on its success.

Checklists are considered to be good options for capturing and packaging experience in software and engineering companies. In this master thesis, we aim to show how experience knowledge can be collected and managed by using checklist.

To conduct our research, an electronic checklist system has been developed and evaluated in an engineering company, Sweden.

AUTHOR KEYWORDS

Experience knowledge, Electronic Checklist, Tacit Knowledge, Knowledge Management.

1. INTRODUCTION

Every company has its own experiences and knowledge that is generated by its employees within regular operations. The level that a company persists and distributes its experience and knowledge among all employees has an important effect on the success in long term. Hence, knowledge is an important asset for a company and it must be effectively managed.

Knowledge Management (KM) and Knowledge Management Systems (KMS) have been studied for several years yet they have gained more interest in recent years. According to [9], KM implementation and use has grown rapidly which is supported by the fact that 80% of the largest global corporations now have KM projects and over 40% of Fortune 1000 companies have a Chief Knowledge Officer, who is a senior-level executive who creates an infrastructure and cultural environment for KM .

Kokkonen takes the researches on knowledge management to one step further and brought a new concept in to the literature named “Experience Knowledge (EK)” which he defines as the knowledge that derives from individuals’ experiences and is persisted in some medium. Experience knowledge, in his definition, is a form of explicit knowledge and the main feature of it is to “formally capture human experiences and enable learning based on them” [10].

There are several mediums that are used to capture and store experience knowledge some of which are; guidelines, reports and checklists. All of them embody some kind of experience and used to transfer that knowledge to others. We, in our master thesis, focus on checklists as an experience knowledge collecting and sharing utility.

Checklists have been widely used to verify the correctness of software documentation and software code. Gilb and Graham (1994) defined them as "a specialized set of questions designed to help checkers find more defects, and in particular, more significant defects". [8]

Although, the main purpose of checklist is to aid in inspection and verification, recently, they are also considered as “experience knowledge collecting and sharing tool” [1].

The relationship between experience knowledge and the generating/using checklist is visualized in figure1. At first, in order to generate checklists, one needs to collect experience knowledge. Then the checklists are executed (used) and during the execution of checklist, more experience knowledge can be collected. Based on these data, the quality of the checklist can be improved. This is the basic circle which is repeated continuously.

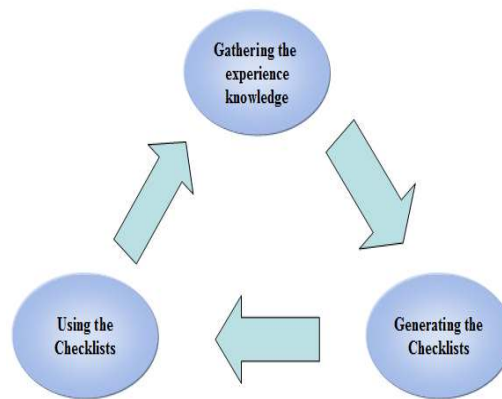


Figure 1.1 Lifecycle of generating/using checklists

Kokkonen argues and shows how checklists act as an experience knowledge collecting tool. In our thesis, we aim to carry this research further and focus on electronic checklist systems with regard to their affect on experience knowledge collecting and sharing.

The research question addressed in this master thesis is: *“In what ways is it possible to manage experience knowledge with checklists? How should a potential checklist system be designed to suit an engineering firm?”*

This thesis is structured into six major sections:

- This section introduces the field of the study, the problem area and the research question that will be investigated.
- Section 2 provides a concise assessment of the previous research work in the field related to this master thesis.
- Section 3 describes the methodology that will be used to answer the research question.
- Section 4 explains the case study conducted in an engineering company.
- Section 5 introduces the electronic checklist system and its properties.
- Section 6 presents the evaluation of results and discussions around applicability of results on software development.

2. RELATED WORK

2.1 Knowledge Management (KM)

As we stated earlier, knowledge management has been a significant issue for decades. The most accepted definition for KM is given in [9] as “the ability to create, validate, distribute, review, share and apply knowledge”. Within researches related to knowledge and knowledge management in organizations, the knowledge as a concept has been also studied and two different form of knowledge have been aroused; tacit knowledge and explicit knowledge [11].

Alavi and Leidne [11] explain tacit knowledge as to be comprised of both cognitive and technical elements whereas explicit knowledge is articulated, codified, and communicated in symbolic form and/or natural language. Hence the distinction between tacit and explicit knowledge is that tacit knowledge is opaque and resides in mind and explicit knowledge is persisted and can easily be transferred to others.

In general, experiences are indicated as tacit knowledge and experience knowledge as explicit knowledge. To convert experiences (tacit knowledge) into experience knowledge (explicit knowledge), there are three phases, and these phases have been named ‘recognition’, ‘preservation’, and ‘exploitation’. In ‘recognition’ phase, a person perceives and understands possible purposes or values of a particular experience, but it is not necessary to make note of the experience at this stage because it still needs to be evaluated. In the ‘preservation’ phase, person change the experience into any kind of form of experience knowledge like training new workers. But this way still has some problem. In the ‘exploitation phase’, experiences are preserved in a solid form that is in public use. One possible solution in general is to use checklist.

2.2 Experience Knowledge and Checklists

Later, Kokkonen put another concept into the field; Experience Knowledge. Experience knowledge, in his definition, “is based mainly on human experiences - with experience defined as <an event or occurrence which leaves an impression on one>” [1]. As seen from his definition, Kokkonen focuses more on “experience” aspect of the knowledge. In his research in [1], he continues; “a particular experience changes from the form of experience to that of experience knowledge having been recognized, preserved and exploited systematically.”

Kokkonen also discussed using checklist to collect Experience Knowledge. He conducted several checklist-generating workshops and concluded that “the checklist generating workshops can be a good practice of collecting experiences and that checklists can work as an experience package.” [2]

In [2], “Experiences from Generating Checklists”, Kokkonen states that checklist is used to improve the quality improvement processes, and he also mentions about several approaches to create the checklist, but his research doesn’t include any solution for maintaining and handling the data that is generated by the usage of checklist.

2.3 Approaches to generate checklist[2]

In [2], “Experiences from Generating Checklists”, three approaches to generate checklists are presented as follows: literature-adopting approach, consultant-based approach and workshop-based approach.

The literature-adopting approach is that the checklists were copied from the literature and slightly modified before use, but it doesn't work because the software development model of the company was usually focused on very different details than the checklists that were adapted from the literature.

The consultant-based approach is that the checklists were generated by a person who knew the software development model and practices of the company. But it also doesn't work, because many of the checklists generated had to be regenerated. And if the consultant who generated the checklist left the company and there is a need to change the checklist, for this kind of case this approach doesn't work.

Workshop-based approach is the best way so far according to [2]. The workshop-based approach was originally based on three phases: the interview, the preparation, and the workshop. Through this process company can involve many workers to generate the checklist; especially company can collect the experience knowledge from different organization and people.

2.4 Checklist usage in engineering companies

A checklist is a tool commonly found in lean product development and is used to assess a technology's maturity, detail design guidelines in every engineering function areas. Almost all well-known engineering companies make use of them.

Checklists are used mainly for assessing a design, a product or a process. However, as Kokkonen and we propose, there also exist companies among engineering companies that explicitly use checklists to capture and transfer experience knowledge.

Toyota exposes one of the best examples of using checklist in engineering production. Toyota is known to be founder of "lean manufacturing" which focuses on efficiency. Instead of long formal documentations, they maintain separate checklist as guidelines for each part in a vehicle. Every engineering function group maintains checklists that detail design guidelines in any number of areas. Checklists contain current capabilities as understood by the responsible designers [14]. Between different engineering functions such as design or body engineering, a checklist is prepared and delivered to the next phase as a guideline package. An engineer in Toyota describes this process with the phrase; "we hand over our experience" [14]. Checklists are followed through the vehicle development process to ensure that all the criteria are satisfied. If a design conforms to its checklist, then the part almost certainly meets a certain level of functionality, manufacturability and quality [14].

A sample excel based checklist used in Toyota is given in figure 2.4.1

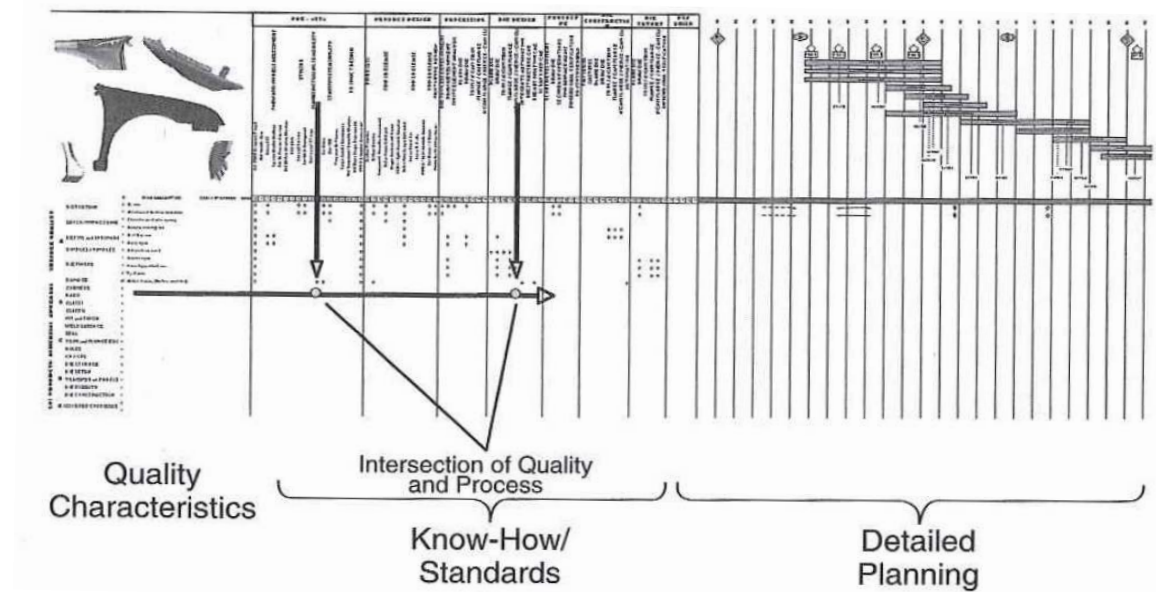


Figure 2.1. Example of checklist from Toyota (excel based, component designer checklist)[15]

2.5 Conclusion for related work

In several research articles, Kokkonen stated and proved that checklist is widely used and important to collect experience knowledge, and he also mentions about several approaches to create the checklist, but his research doesn't include any solution for maintaining and handling the data that is generated by the usage of checklist.

In the current situation, companies are often using excel based checklist. Storing the checklist as excel files and distributing them by printing satisfies maintainability if the company is small and if there is only a small number of checklists in the company. However, for big companies which utilizes big amount of checklist and when a checklist is used among several departments, excel based checklist systems are simply not enough for successful management. With excel based checklist, it's hard to control huge data and there is a big challenge to maintain these checklist with excel model.

3. Methodology

To conduct this research study, literature review and empirical research methods are used.

At first, the literature is scanned to clearly see the relation between checklist usage and knowledge management in industrial companies. We have presented the findings of this review in Section 2, related works.

The empirical part of our research comprised of case study in our case company. After the literature review, we went on the development of our proposed system. Our aim was to find out real benefit of checklist system, especially electronic checklist system, as a knowledge collection tool in a product development company.

3.1 Development of the system

We interviewed 2 scientists currently involved in the Technology development process at case company, in order to get the specific requirements of the company. To be accepted by the case company, our system had to be developed in a way that satisfy their needs and support their work structure. That brought us new requirements such as project support which is not related to checklists at core.

As for any research based software development project, the main obstacle for us was that there was no example of our proposed system. Neither our customers had a clear idea of final system. Together with the interviewed customers, we had discussions on requirements, features and user interface, thus gradually developing the system.

Since the requirements and supported features were so opaque and we worked as a very small team of two master students, we followed agile techniques. We defined work packages in our periodic meetings, implemented the planned features, than reviewed and discussed again about next features to be implemented. Thus, we developed the entire system in varied length iterations.

At first, the system was very imaginary for all of us and we got difficulty having the same understanding about it. We utilized prototypes for the first phases to have same understanding and to save time. Later in the development, we had a base implementation and continued with real system instead of prototypes.

We developed the system in ASP.NET (C#), Microsoft Sql Server. Since, our university has agreement with Microsoft, we had not any licensing problem and since we had previous experience in asp.net, we quickly started development without a learning phase. The activity diagram presented in Figure 5.1 shows the execution process followed in order to accomplish our study.

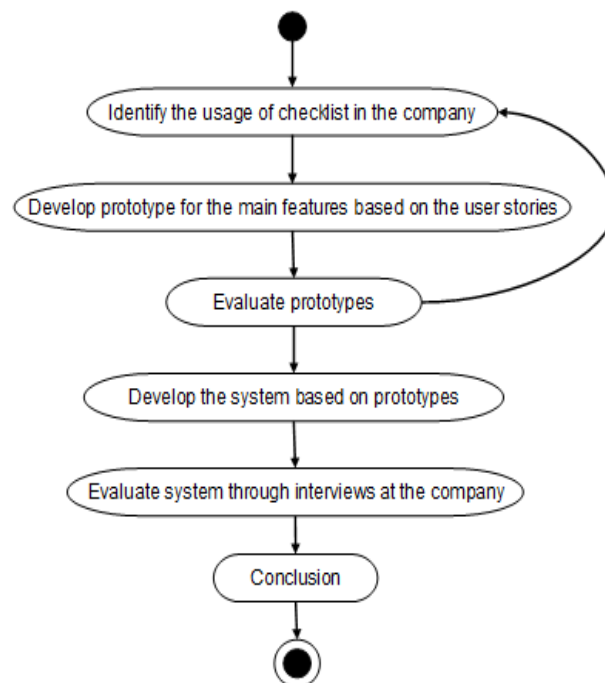


Figure 3.1: Execution process of the case study

Our case study is divided into 4 phases;

Phase 1: identify the usage of checklist in the company.

In this phase we tried to understand; various types of checklists that are used in companies and differences between them, how the checklist is used, what do they want to do with checklist. To get this information we had interviews with the company representatives and collected some user stories. During the interview we got a general understanding of checklist usage in the case company there are two types of checklist, process checklist and product development / method checklist.

The detail information is shown in Table 3.1.

Type of checklist	Usage of checklist
Process Checklist	A checklist that is assigned to a project e.g. to assure that gates are passed or that the product or technology is mature
Method Checklist	A checklist that is owned by a person (normally a method owner) in order to make sure of correct execution of a method.

Table 3.1 Types of checklist and the usage

Phase 2: Develop prototype for main features based on the user stories.

After identifying the usage of checklist in the phase 1, we developed few prototypes to make sure that all of us understand the same thing. While some of the prototypes were simple drawings, some other were developed using ASP.NET to show our customers the real limitations and possibilities of real development environment.

Phase 3: Evaluate prototypes and develop the system

In this phase we had meeting with end users by showing our prototypes, and we got feedback from them, so we can clarify the requirements more clearly. The evaluation meetings also helped the end users to understand the new system thus we could have better suggestion from them.

For both the development of prototypes and real system, iterative process is used which allowed us to make modifications and improvements of the prototypes dynamically, and have more clear requirements from the users. Customers also had more clear understanding of how the process can be reflected on the system.

Phase 4: Evaluation of the system

Finally, when the system became mature enough to be presented to other employees in case company, our interviewed representatives arranged some workshops with few project managers and method owners. Few realistic scenarios have been setup for the workshop and an evaluation form was prepared to get their feedback. The evaluation result is presented in following section.

4. Case Study Design

As a part of our research, we performed a case study in an engineering company, Volvo Aero Corporation (VAC), in Trollhättan. The case company mainly develops components for commercial jet engines. To secure their leading position in industry they invest in new technology development, and also continuously develops their methods for performing technology development. The process used for technology development is shown in Figure 4.1. The case company is suitable for this research work since they consider checklists as a part of their quality assurance process in technology development. This means that checklists could be applied both to the toll gates (*process checklists*) and to assure the quality of the methods and technologies developed (*method checklists*).

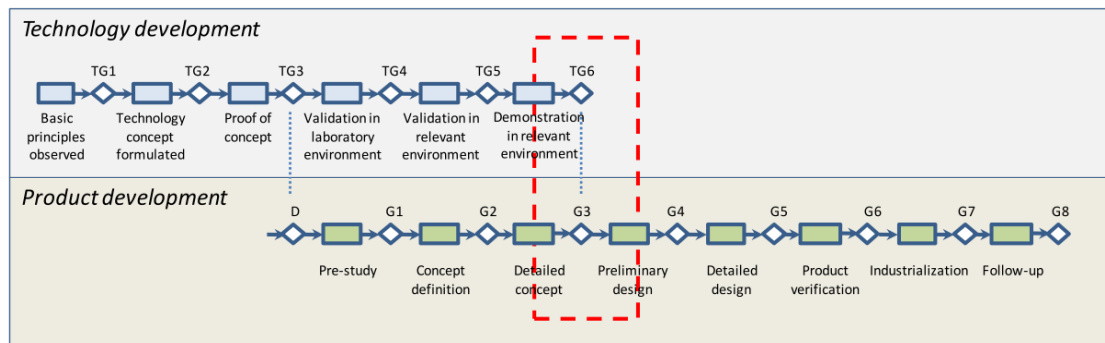


Figure 4.1 A typical gated processes for technology and product development [16]

The case company follows gated process for its technology development projects. These projects are related to Technology Readiness Level (TRL) and when passing from one TRL level to another there is a corresponding checklist for that TRL level. In figure 3.1 The TG1-TG6 relates to the TRL levels TRL1-TRL6. The TRL checklists are based on the general NASA and DoD Checklist [13]. Apart from TRL checklist there is a potential for utilizing other types of checklists for example for personal use or for product quality assurance as well as for ensuring correctly applying invented technologies.

Although, the TRL checklists are utilized today, the current file based system does not allow continuous improvements as a potential dynamic system would. However, experience from Japanese companies show that checklists used in product development, can be maintained and improved continuously.

Our focus in this case study is on utilizing checklist in design process to help designers to follow related methods in their design, thus focusing on method checklists. Checklists that can be defined by method owners and executed/used by product designers are named as *method checklist*. These checklists could be regarded as guidelines, created by a technology developer and followed by designers when designing a new product or preparing it for

production. After creation, as the technology or method evolves, the checklist can be updated and improved either by the method owner or by a designer using the checklist.

Creation of method checklists is an important process of converting tacit knowledge into experience knowledge. When a technology developer designs a new method for doing something, he/she prepare a checklist as a guideline and capture the knowledge in explicit form. Therefore the knowledge that a technology developer has is converted to experience knowledge. However technology developer is not the only person who knows about the new technology. Product designers also utilize and gain experience of using that technology in tacit form. A good knowledge management system must also collect knowledge from product designers by producing experience knowledge from that.

Product designers or other employees may have tacit knowledge not only regarding existing checklist but also other daily operations. That tacit knowledge could be tips and tricks about daily operations or things to be careful at.

Based on these facts an electronic checklist system is developed. However, the initial requirements when we start our development were different. Following user requirements were created at the beginning of the project:

- 1) A manager should be able to investigate maturity of a technology.

A manager wants to see the status of projects to know when they can be ready for the production.

- 2) User should be able to give comment to every check items.

If there is a problem with check items which prevents assigning a value, user can give comment and skip it, so next time the user of check list can see it.

- 3) User should be able to change (add, modify, remove) the check items, and it should be reflected to all the check lists which are using same check items, but before it get approval, it should be marked.

This story is for the case that a user of a checklist finds a problem with a check item, then he should be able to change it with giving a proper comments. But before the change gets approval from the checklist owner, it will be displayed with distinguished marker.

- 4) Same checklist shall be used in several projects, without value conflict.

When a new project start, the method owner find that the same checklist which is used in the previous project can be used in this new project, so he can just find it from the system and assign it for this new project and use it.

- 5) To be able to let a project pass to next gate, it should be approved by the method owner.

Method owners must review and approve the checklist execution result when passing to new gate.

The resulting system provides most of the requirements, but in a different way. Few screenshots of resulting system are given below.

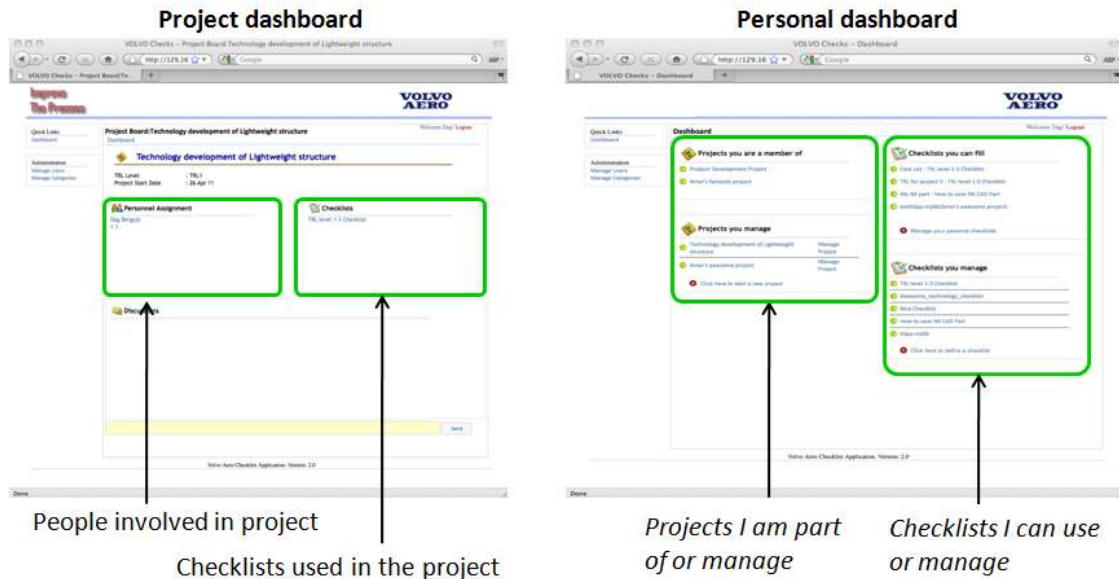


Figure 4.2. Screenshot of dashboard pages of system.

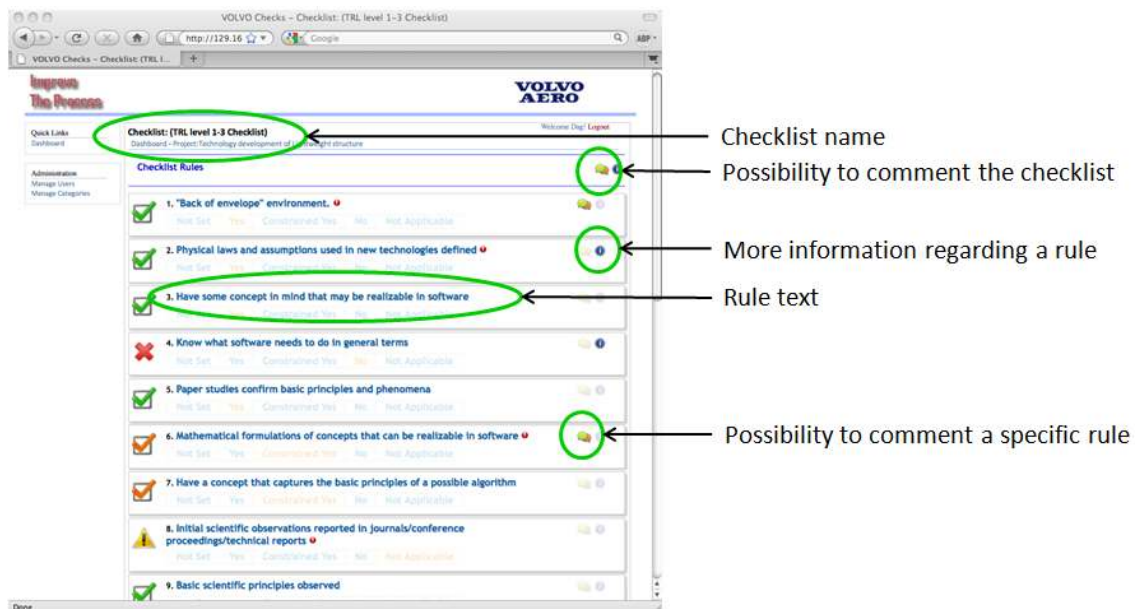


Figure 4.3 Easy to use interface for using checklists

5. Electronic checklist system

Kokkonen states that “checklist generating workshops can be a good practice of collecting experiences” [1]. In our research, we want to prove and expand this statement to that “not only checklist generating workshops but also checklist usage can be a practice of collecting experiences”. We want to show that experience knowledge in a product development company can be managed by some sort of checklist system.

In order to achieve our goal, we developed an electronic checklist system for defining/managing/using the checklists in the company.

Our purpose in developing the system was to have maximum level of collaboration among all employees hence let them input their tacit knowledge. The result is a checklist and project management system. Collaboration is achieved by allowing users to discuss around projects/checklists or even check items (rules). Product designers (checklist users) can even contribute to checklists definition by modifying or adding new check items to the checklists. Therefore, our system allows continues development of checklist and continues collecting of experience knowledge from product designers.

Figure 5.1 depicts the system. In general, our system acts as a checklist management system and a checklist repository. Instances of checklists can be used in any project or even for individual usage.



Figure 5.1 Overview of the system usage

5.1 Collaboration via electronic checklist system

In the previous section, we mentioned about approaches to generate the checklist and we concluded that the workshop-based approach is the best way, in other words, collaboration between the workers is the most important way of collecting the experience knowledge and generating the checklist. Therefore, our main goal in developing this system was to have collaboration.

Project users can have chat over check items or checklist (Figure 5.2). They can add new check items or modify existing check items with proper comments (Figure 5.3). Based on configuration, a modification of a check item can be directly visible to other projects that are using the same checklist.

With this feature, we believe that our system provides a very effective “knowledge collecting” ability since the checklist will be gradually developed by all employees in the company.

16. Know what program the technology will support
☐ Not Set ☐ Yes ☐ Constrained Yes ☐ No ☐ Not Applicable

17. An apparent theoretical or empirical design solution identified
☐ Not Set ☐ Yes ☐ Constrained Yes ☐ No ☐ Not Applicable

18. Basic elements of technology have been identified
☐ Not Set ☐ Yes ☐ Constrained Yes ☐ No ☐ Not Applicable

19. Desktop environment
☒ Not Set ☐ Yes ☐ Constrained Yes ☐ No ☐ Not Applicable
 Dag Bergsjø 17 Jun 11 16:00 It is not good enough

20. Components of technology have been partially characterized
☒ Not Set ☐ Yes ☐ Constrained Yes ☐ No ☐ Not Applicable

Checklist Discussions
 Dag Bergsjø 17 Jun 11 15:59 It is good TRL checklist

Figure 5.2 Comments for checklist and check items

5.2 Checklist management aspect of electronic checklist system

One of the main problems with excel based checklist system is to track the projects in which the checklist is used and evaluate execution result from several projects. In our proposed system, a checklist is defined once and can be used several times in any project. Project managers can choose which checklists to be used in their projects from checklist repository and checklist owners can follow usage results of their checklists.

System tracks changes of check rule definitions as well as change of check rule values.

Modify Checklist Rule

Rule: Desktop environment

Rule Description :

Change Comment :

Dag Bergsjø 17 Jun 11 16:00

Figure 5.3 Check items' modification feature with comments

6. Evaluation and Discussions

A small group of people mainly from managerial level have been interviewed in the company for the usage of electronic checklist system. They were project managers, method and process owners.

Several aspects of the system have been discussed in the interviews and the evaluation as well as discussion about improvements is given in following sub sections.

6.1 Reactions on “electronic checklist system”

Currently, the case company documents their checklists in word files as templates and stores those files in SharePoint portal. When a checklist is needed to be filled in a process or in a gate pass review, that checklist is copied, filled and stored again in the portal. If a discussion is needed for filling the checklist, they have a meeting to discuss it and submit only the result into the portal. Therefore, the knowledge which appears in the discussion cannot be captured and remains as “tacit”.

The SharePoint aided checklist system of the case company also lacks a collaborative environment. As we stated earlier, updating and enhancing the checklists are of checklist owners’ responsibility. Since the checklist owner (method/process owner) and the users of checklist (designers) are from different departments, it is almost impossible for checklist owner to benefit from others opinion. This was the strongest part of our system. No matter the checklist user is in which department or in which project, if checklist owner allows via configuration, they can modify or comment on check rules. Thus checklist owner can benefit the most from opinion of all related people in the company and shape the checklist in the best possible way.

Our system also has advantages from a checklist management perspective. Currently, a checklist owner cannot know which projects are using his/her checklist. That is an important need since when an update is needed for a checklist, its copy in any project could become outdated. All the checklist and check items are stored and used online in our system. A checklist owner can track the projects that are using his/her checklist, and when a checklist is updated, the change is reflected to everywhere at the same time.

The electronic checklist system is very configurable via management pages. For the demonstration, a very flexible scenario has been established in which almost every checklists can be modified regarding their check items by its users. The first reaction of interviewees for this flexibility was that they get anxious about authorization. Since they were from managerial level, they are most interested in who can do what, and their first opinion was that it is not good to let everyone to change check rules. However, when they see the configuration options, they accepted the system and their worries were satisfied.

6.2 Reactions on “work procedure in electronic checklist system”

Although there are different employee types in the case company, we approached all of the employees same, that means there is no designer, manager, checklist owner distinction in our system. Anyone can create checklist or project in the system and become checklist owner or project owner. After that, those owners can modify as they wish. Therefore there is not a strict map between company structure and our system regarding employees and rights.

When a checklist is created, the checklist owner chooses whether the checklist is published or remains private. All the published checklists are stored in the checklist repository and any project manager can add them into his/her project. Private checklists are not visible to anyone except its creator. They can be considered as self-notes. This also allows gradual development of checklists and checklist owner can publish the checklist when it is ready. An additional benefit of “private checklist” feature is that, the knowledge packaged in that private checklist can be easily transferred to new employees.

Changing check items by users, which is the most significant feature, is fully configurable in our system. A checklist owner can select/unselect following options when defining the checklist; users can change check items / users can add new check items / check item changes can be visible to other projects before approval. Thus, for a TRL like checklist which rarely changes, checklist owner can disable modifications by users and for a design checklist which evolves by the experience, designers can be allowed freely update the checklist content. System contains a page to track the changes, related comments and options to approve/reject the change. Unfortunately, this feature is not assessed by the interviewees at the desired level. However neither we had a bad feedback about this feature and that led us to think that the work procedure of our system satisfies their needs and is better than their current system.

6.3 Potential future development

We should accept that the implemented system is not fully inspected by the interviewed employees in the company. The reason was that they expected a diverse solution that also manages their projects and maybe their documents. They also expected to have more graphics in the system. Therefore, we think that in order a system to be accepted by a company, it must be fully integrated with their other systems and their work structure as well and it must be user friendly.

The interviewed managers gave us many suggestions regarding additional features.

Access to checklist system via mobile devices is the most probable extension to the system. That will probably increase checklist usage in the company thus more experience knowledge can be captured. Considering that many people have touch enabled phones, accessing the system via mobile devices will attract employees interest for using system.

Graphical support for project managers is the second most suggested feature. The interviewed managers wanted to use the system as a decision support system and see the status of developed technologies graphically as an aid in their decisions.

Supporting checklist statuses with graphics is also a needed extension. Currently system shows a percentage for interpreting the status of a checklist. When all of check items has been set to “yes”, it has a 100% value. This value can easily be converted to graphics thus it can be easier to perceive the status for project owners.

6.4 Applying checklist system to software development companies

Although our system is customized for case company, which is a technology development company, it can easily fit into software development companies.

Checklists are widely used in software companies in quality assurance process. Inspections are commonly performed with the help of checklists and software companies usually utilize other type of checklist for ensuring their organizational rules to be followed.

One question here could be that “how much tacit knowledge is there in the employees’ mind, and how much of it can be converted to experience knowledge (transferrable form)?” Despite the situation in case company’s design checklists, inspection checklists in software companies are more stable. That means the checklist system will probably not as effective as it is in engineering companies. However, it is still possible to use it as personal documentation tool as it allow any user to create their own checklists.

Additionally, as a checklist management system, without benefiting much of experience collection aspect of it, it can be still much valuable for the software development companies.

7. Conclusion

This master thesis presents way of how can we collect and manage the experience knowledge in an efficient way and generate/use the checklists to improve the process in engineering companies with a case study.

The evaluation shows that a well implemented checklist system can be used as a part of Knowledge Management System in engineering companies in order to collect tacit knowledge and capture it into form of checklists. Not only checklist generation phase but also checklist usage phase can be considered as a process of collecting tacit knowledge by utilizing electronic checklist system.

In our research we supported and extended the finding of Kokkonen which is “checklist generating workshops can be a good practice of collecting experiences” by showing that it is also possible to collect experience while using checklist.

We have also faced with the difficulties that are common to new software products related to getting acceptance by its users. In order to widely accepted in the company and widely used by users in the company, a software product must serve to diverse requirements in the company rather than solving a specific problem which in our case is collecting tacit knowledge.

Visuality is also important for software products to be used by large number of people. People are eager to use visually appealing products and if your product is not visually appealing, even if you convince the company at managerial level, you still have difficulties in having users effectively use the system.

ACKNOWLEDGMENTS

The authors are grateful to all managers and contact person at the company for their support and contribution during this research. Special thanks to the supervisors and interviewees at the company who participated in the interviews for their helpful comments and suggestions.

Reference

- [1] J. Kokkonen, Checklists as a Tool for Collecting Experience Knowledge, *Proceedings of IASTED International Conference of Information and Knowledge Sharing*, St. Thomas, USVI, 2002, 223 - 228.
- [2] Kokkonen, J., "Experiences from Generating Checklists". Proceedings of the IASTED International Conference on Knowledge Sharing and Collaborative Engineering (KSCE2006), St. Thomas, Virgin Islands, USA, 2006, pp. 51–56.
- [3] Birk, A. and Tauz, C., "Knowledge Management of Software Engineering Lessons Learned", Proceedings of the 12th Conference on Software Engineering and Knowledge Engineering, 1998, pp. 24–31.
- [4] Basili, V., Caldiera, G., and Rombach, H.D., "The Experience Factory", Encyclopedia of Software Engineering, vol 1, John Wiley & Sons, 1994, pp. 469–476.
- [5] Nonaka, L. and Takeuchi, H., *The Knowledge-Creating Company – How Japanese Companies Create the Dynamics of Innovation*, Oxford Press, 1995.
- [6] The Concise Oxford Dictionary, tenth edition (Oxford University Press Inc., New York, 1999).
- [7] Kokkonen, "Gathering Experience Knowledge from Iterative Software Development Processes"
- [8] Kokkonen, "The Lifecycle of Checklists: A Knowledge Management-based approach", Proceedings of Third European Conference on Knowledge Management, Dublin, Ireland, 2002, 376-386.
- [9] Zurita, G., Balloian, N., and Baytelman, B., "Mobile Collaborative Knowledge Management System"
- [10] Kokkonen, "From Software Documents to Experience Knowledge Based Artifacts"
- [11] Alavi, Maryam; Leidner, Dorothy E. "Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues", 2001, *MIS Quarterly* **25** (1): 107–136.
- [12] Stig, D. C., and Bergsjö, D., "Means for internal knowledge reuse in predevelopment – the technology platform approach", 2011
- [13] William L. N., Brian C. K., and Roger J. D., "Technology Readiness Calculator",
- [14] Durward K. S., Allen C. W., Jeffrey K. L., "Toyota's Principles of Set-Based Concurrent Engineering", 1999
- [15] Morgan, J. M. and Liker, J. K. "The Toyota Product Development System", 2006, Productivity Press, New York, USA

[16] Högman, U., Bengtsson, D., and Stetz, S., “Requirements on New Technology and the Technology Implementation Process”, 2010

Appendix A – Questionnaire for evaluating system

Evaluation form for “VOLVO CHECKS”

Name, Last name:

Role in the company:

For those who involved in checklist definition/maintenance

1. Are you involved in maintenance of any checklist?

If yes please list their names and your role:
2. How those checklists are first defined? What are their information source?
3. Are your checklists used in more than one department/project? If it is so, how do you manage sharing them between multiple departments?
4. How frequently are checklists updated and why are they updated?
5. How important is the opinion/feedback of checklist users (those who fills the checklist) about checklists they are filling? Please give your reason.
6. How much degree that the feedback from checklist users are collected and reflected to checklists; and do you think that it is enough? Please give your reason.
7. What do you think in general about “Volvo Checks” application in terms of checklist management?

For regular users who fill checklists

1. How many projects are you involved in and how many different checklists are being used in those projects?
2. Do you think that at least some of the check items/rules in those checklists can be improved?
3. What do you do when you find a problem in the checklist to correct it?
4. Do you fill the checklists for any project individually or with your team mates?
5. How do you achieve collaboration when a checklist needs to be filled by a group?

6. Are you taking notes as a “guidance” in order to document your common tasks? If so, how easy is it to manage/maintain them?
7. Have you ever wanted to define your own checklist for any reason (for personal usage or sharing with others)?

Questionnaire for everyone

Please answer following questions by assigning a point to each one.

(1-very difficult 2-difficult 3-normal 4-easy 5-very easy)

		1	2	3	4	5
1	How easy is it to use electronic checklist system ?					
2	How easy is it to use excel based checklist ?					
3	How easy is it to manage/maintain the checklist by using the electronic checklist ?					
4	How easy is it to manage/maintain the checklist by using the excel based checklist ?					
5	How easy is it to share the data with different departments by using the electronic checklist ?					
6	How easy is it to share the data with different departments by using the excel based checklist ?					
7	How easy is it to collaborate with your group members by using electronic checklist system ?					
8	How easy is it to collaborate with your group members by using excel based checklist ?					
9	Do you like to use electronic checklist system instead of using the excel based checklist?					

Thank you for sharing your idea with us